

USO DE DATOS



CONSTANTES

Una **constante** es un dato cuyo valor no puede cambiar durante la ejecución del algoritmo.

Representan una dirección de memoria en donde se almacena un dato, y su contenido **puede ser modificado** en la ejecución del algoritmo cuantas veces sea necesario.

Lo tengo que usar cuando tengo un valor definido y estático varias veces dentro de un algoritmo.

Recuerda que ese dato ocupa espacio en memoria, por lo que no debemos definir constantes por doquier sin analizar realmente la necesidad de hacerlo

VARIABLES

Una **variable** es un espacio de memoria donde se almacenan datos que pueden ser modificados durante la ejecución del algoritmo.

Numérico

- Enteros
 - *Por ejemplo edad, cantidad de hijos*
- Reales
 - *Por ejemplo sueldo, peso, altura*

Alfanumérico

- *Por ejemplo nombre, nacionalidad*

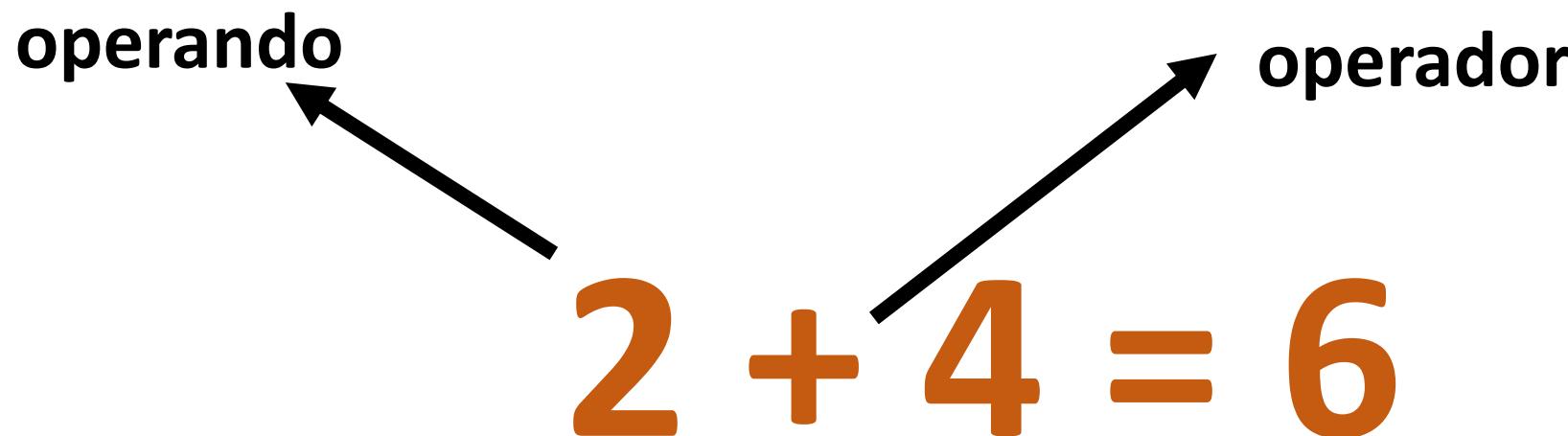
Lógico

- *Por ejemplo si es socio o no*



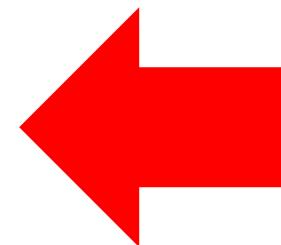
TIPOS DE DATOS OPERADORES

Un **operador** es el símbolo que determina el **tipo de operación o relación** que habrá de establecerse entre los **operandos** para alcanzar un **resultado**.


$$2 + 4 = 6$$

Operador	Descripción
+	Suma
-	Resta
*	Multiplicación
/	División real
MOD	Resto de la División
DIV	Cociente de la División

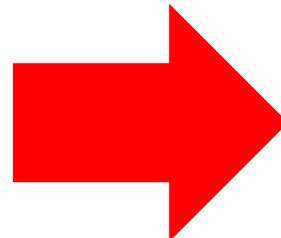
OPERADORES ARITMÉTICOS



Son **operadores binarios** (requieren siempre dos operandos) que realizan las **operaciones aritméticas** habituales.

OPERADORES RELACIONALES

Sirven para **realizar comparaciones** de igualdad, desigualdad y relación de menor o mayor. Estos operadores sirven para **expresar las condiciones** en los algoritmos.



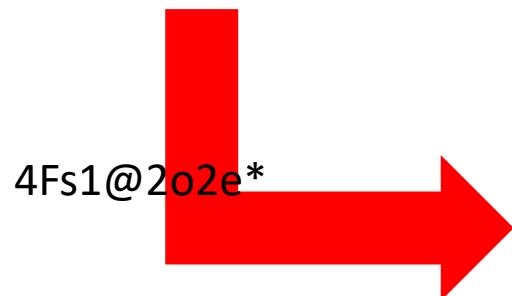
Operador	Significado
=	Igual a
\neq	No igual a
>	Mayor que
\geq	Mayor o igual que
<	Menor que
\leq	Menor o igual que

Proporcionan resultados lógicos.

OPERADORES LÓGICOS

Son aquéllos que permiten la **combinación de condiciones** para formar una sola **expresión lógica**

Utilizan operandos lógicos y proporcionan resultados lógicos también.



Operador	Descripción
Y / AND	Operador lógico AND
O / OR	Operador lógico OR

A Y B			A O B		
A	B	RESULTADO	A	B	RESULTADO
V	V	V	V	V	V
V	F	F	V	F	V
F	V	F	F	V	V
F	F	F	F	F	F

¿Qué son las ESTRUCTURAS de CONTROL?

***Se utilizan para controlar
la ejecución de las acciones
dentro del algoritmo. El
control puede ser
secuencial, condicional o
iterativo.***

ESTRUCTURAS SECUENCIALES

```
1  LEER(A);
2
3  LEER(B);
4
5  SUMA := A + B;
6
7  ESCRIBIR('LA SUMA ES = ', SUMA);
```

ESTRUCTURA CONDICIONAL SIMPLE

```
1 SI [CONDICION] ENTONCES
2
3     [ACCIONES] <-- Pueden ser simples o complejas, una o varias...
4
5 FIN_SI;
```

ESTRUCTURA CONDICIONAL ALTERNATIVA

```
1 SI [CONDICION] ENTONCES
2
3     [ACCIONES] <-- Pueden ser simples o complejas, una o varias...
4
5 CONTRARIO
6
7     [ACCIONES] <-- (Ídem anterior)
8
9 FIN_SI;
```

ESTRUCTURA

CONDICIONALES MÚLTIPLES

```
1 SEGUN [VALOR] HACER
2
3     VAL1: [ACCIONES] <-- Pueden ser simples o complejas, una o varias...
4
5     VAL2: [ACCIONES] <-- (Ídem anterior)
6
7     VAL3..VAL6: [ACCIONES] <-- (Ídem anterior)
8
9     VAL7, VAL9, VAL11: [ACCIONES] <-- (Ídem anterior)
10
11 CONTRARIO
12
13     [ACCIONES] <-- (Ídem anterior)
14
15 FIN_SEGUN;
```

ESTRUCTURAS REPETITIVAS

Permiten **repetir** la ejecución de una o varias acciones un **número determinado de veces**.

ESTRUCTURA PRE-TEST

```
1 MIENTRAS [CONDICION] HACER
2
3     [ACCIONES] <-- Pueden ser simples o complejas, una o varias...
4
5 FIN_MIENTRAS;
```

ESTRUCTURA POST-TEST

```
1 REPETIR
2
3   [ACCIONES] <-- Pueden ser simples o complejas, una o varias...
4
5 HASTA QUE [CONDICION];
```

ESTRUCTURA MANEJADA POR CONTADOR

```
1 PARA VC := VI HASTA VF, INC (o DEC) HACER
2
3 [ACCIONES] <-- Pueden ser simples o complejas, una o varias...
4
5 FIN_PARA;
```



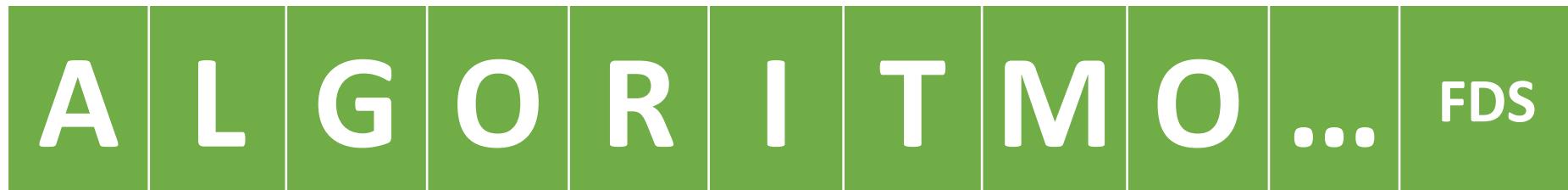
ALGORITMOS Y ESTRUCTURAS DE DATOS

UNIDAD II: Secuencias de Datos Elementales



QUÉ ES UNA SECUENCIA

un conjunto de datos relacionados entre sí que deben cumplir con ciertas características



1 2 3 n

SECUENCIA Características

✓ Primer objeto de la secuencia

El acceso a este elemento permite el acceso a los demás

✓ Relación de sucesión entre los objetos

Todo elemento (excepto el último) precede a otro.

Todo elemento (excepto el primero) es el sucesor de otro.

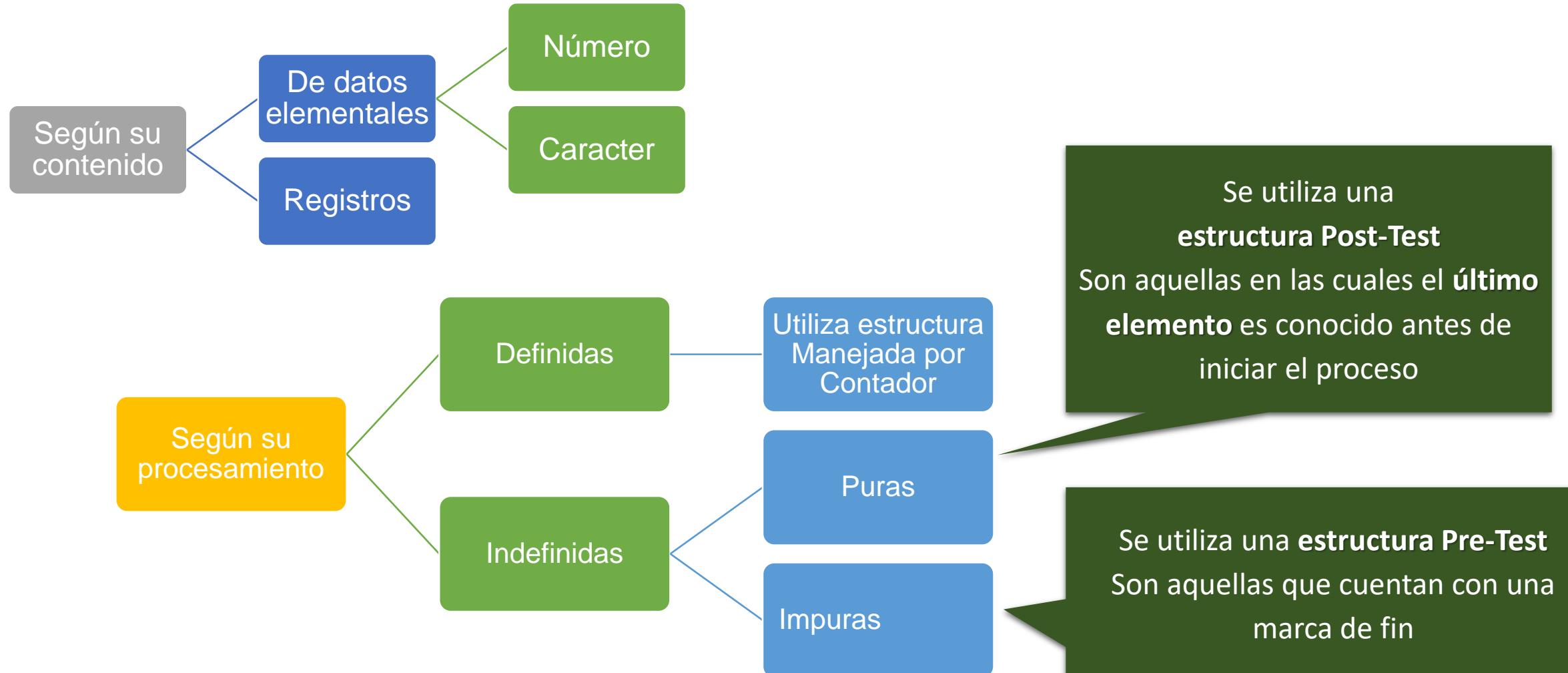
✓ Existencia del último elemento de la secuencia

Debe existir un último elemento de la secuencia

✓ Caracterización del fin de la secuencia

Debe estar definido un indicador de fin de secuencia

SECUENCIA Clasificación



SECUENCIA Subsecuencias

Conjunto de elementos consecutivos, que están incluidos en la secuencia, considerados como un **sub conjunto** de acuerdo con la definición del problema.

SECUENCIA Subsecuencias

Las sub secuencias también deben cumplir todas las características de una SECUENCIA, como ser:

- **Existencia del primer elemento de la secuencia**
- **Relación de sucesión entre los elementos**
- **Finitud**
- **Existencia del último elemento de la secuencia**

Y también se las clasifica de acuerdo a su contenido y procesamiento.

SECUENCIA Relación entre subsecuencias

Sub secuencias jerárquicas

SECUENCIA PRINCIPAL

ORACION

PALABRA



ALGORITMOS Y ESTRUCTURAS DE DATOS

UNIDAD III: Estructuras – SUB UNIDAD A



QUÉ ES UN REGISTRO

Un registro, en programación o diseño de algoritmos, es un **tipo de dato estructurado** formado por la unión de varios elementos, lógicamente relacionados, bajo una misma estructura.

REGISTRO

CARACTERÍSTICAS

HETEROGENEIDAD DE LOS DATOS

Sus elementos pueden ser de DIFERENTES TIPOS

234

norte

verificado

042018

234	Norte	Verificado	042018
-----	-------	------------	--------

REGISTRO: CAMPOS

- ✓ Un **campo** es la **entidad lógica** más **pequeña** que conforman al registro.
- ✓ Un **campo** es la **unidad mínima** de **información** de un registro (puede ser un dato estructurado...).

REGISTRO

DEFINICIÓN y EJEMPLOS

El registro debe definirse en el ambiente para ser usado por en el algoritmo.

```
ACCION GestiónPersonal ES;  
AMBIENTE  
PERSONA = Registro  
  DNI: Entero;  
  Nombre: AN(50);  
  Domicilio: AN(70);  
Fin Registro;
```

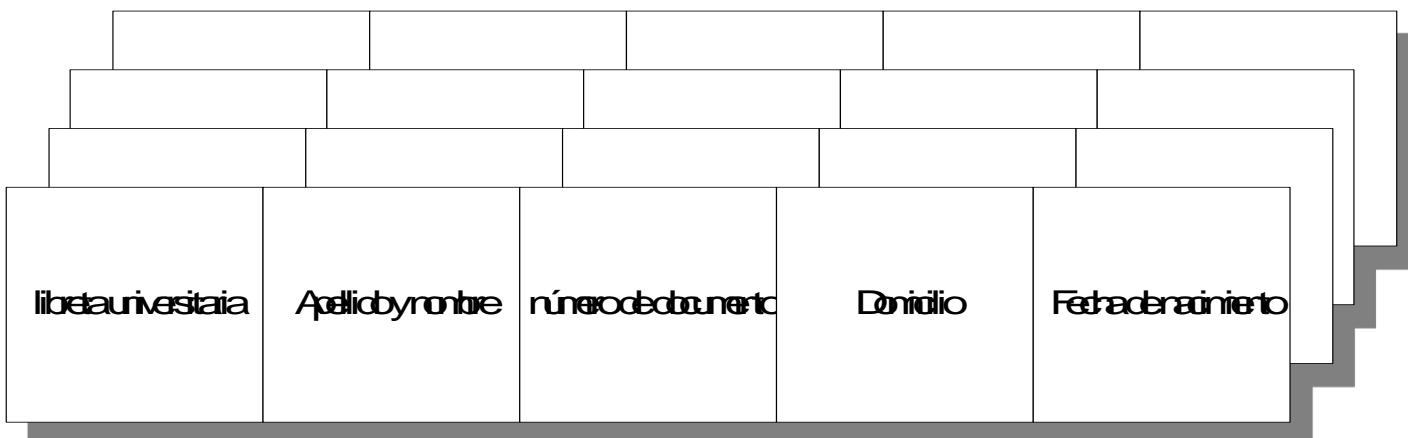


¿Qué es
un
Archivo?



QUÉ ES UN ARCHIVO

Un *fichero* (**archivo**) de datos es una colección homogénea de registros **relacionados** entre sí y **organizados** para un propósito específico.





DATOS INTERESANTES

- Un archivo está siempre almacenado en **memoria externa**, su proceso se realiza en la memoria interna.
 - ✓ La información almacenada es permanente.
- Existe **independencia** de los datos respecto de los algoritmos.



DATOS INTERESANTES

- Todo algoritmo intercambia datos con el archivo y la **unidad básica de entrada/salida es el registro.**
 - ✓ Los datos extraídos o almacenados en el archivo son los de un registro completo.
- Los archivos en memoria externa permiten una **gran capacidad de almacenamiento.**



ORGANIZACIÓN DE ARCHIVOS

Es la manera en que se **encuentran**
almacenados los **registros** en el
archivo.

ARCHIVO: ORGANIZACIÓN

Existen 3 tipos de ORGANIZACIÓN:

1. **SECUENCIAL**
2. **RELATIVA** (o DIRECTA)
3. **INDEXADA** (o SECUENCIAL INDEXADA)

ORGANIZACIÓN SECUENCIAL

Principio del archivo



Registro 1
Registro 2
:
:
:
Registro I – 1
Registro I
Registro I + 1
:
:
:
Registro N – 1
Registro N

Fin del archivo



ORGANIZACIÓN INDEXADA

Área de índices	CLAVE	DIRECCIÓN
	15	010
	24	020
	36	030
	54	040
	.	.
	.	.
	.	.
	240	090

Área principal
↑
(Área de Datos)

CLAVE	DATOS
010	15
011	
012	
.	
.	
019	
020	24
021	
.	
.	
029	
030	36
031	
.	

ORGANIZACIÓN RELATIVA

Almacenado en un soporte **direccional**, es decir que los registros se localizan por su dirección.

Los registros deben contener un campo específico denominado **clave** que **identifica cada registro de modo único**, es decir, dos registros distintos no pueden tener un mismo valor de clave.



ACCESO DE ARCHIVOS

Indica la manera en que se recuperan (o leen) los registros.

ARCHIVO: ACCESO

Existen 3 tipos de ACCESOS:

- 1. SECUENCIAL**
- 2. DIRECTO**
- 3. MIXTO O DINAMICO**

**¿Cómo identifico un
registro en un
archivo?**

Un **campo clave (key)** es aquel que **identifica al registro y lo diferencia de los otros registros.**

(Clave PRIMARIA)

Debe ser **UNICO**, es decir **debe ser diferente para cada registro dentro del mismo archivo.**

Campo Clave Simple

Está formado por un **campo contenido**, es decir, no se encuentra subdividido.

Campo Clave Compleja

Está formado por un **campo continente**, es decir, se encuentra subdividido en campos contenidos.

En estas claves lo que se respeta es el orden de jerarquía de los campos.

CLAVE			DNI	ApeNom	Domicilio
Facultad	Carrera	Legajo			

Clave secundaria

Se denomina así a un campo o a una combinación de campos que permite relacionar el registro con información en el mismo archivo u en otros soportes.

Clave foránea

Es también una clave secundaria, pero implica una relación exclusivamente con otro archivo , es decir permite la relación entre dos estructuras externas. La clave foránea identifica una columna o grupo de columnas en una tabla- archivo (tabla hija o referendo) que se refiere a una columna o grupo de columnas en otra tabla-archivo (tabla maestra o referenciada), normalmente en esta última esa clave es primaria.

CARACTERÍSTICAS DE LOS DATOS ALMACENADOS EN ARCHIVOS

La Consistencia de un archivo es la propiedad que verifica la validez del dato almacenado con su definición en el ambiente.

FECHA = Registro

 DD: Entero;

 MM: Entero;

 AAAA: Entero;

Fin Registro;

Reg = (02/12/2019) Es consistente

Reg = (50/50/9999) Es consistente

Cuando hablamos de **Consistencia Automática** hacemos referencia a la definición de límite para los datos por ejemplo Rango, Conjunto.

FECHA = Registro

 DD: 1..31;

 MM: 1..12;

 AAAA: Entero;

Fin Registro;

La Congruencia de un archivo es la propiedad que verifica la validez de los datos entre sí.

- **Fina:** Validación entre datos en archivos distintos (Datos de un campo, con datos de otro archivo, por ejemplo un DNI en un archivo de ALUMNOS, validado con un archivo de PADRON)
- **Gruesa:** Validación entre datos en un mismo registro (**Por ejemplo:** **día/mes/año -> 31/02/2019 ... sería consistente pero no congruente**)

NOTA1

si es consistente, no indica que sea congruente.

NOTA2

si no es consistente, tampoco es congruente.



UNIVERSIDAD
TECNOLÓGICA NACIONAL
FACULTAD REGIONAL
RESISTENCIA

INGENIERÍA EN
SISTEMAS DE INFORMACIÓN

ALGORITMOS Y ESTRUCTURAS DE DATOS

UNIDAD III: Estructuras – SUB UNIDAD A

PROCESOS con ARCHIVOS

PROCESOS INDIVIDUALES

Un proceso **es individual** cuando existe un único Archivo de Entrada y **1 o ningún Archivo de Salida.**



PROCESO	CARACTERISTICAS
Genérico	Es el proceso de carga o generación , tiene como objetivo crear un archivo consistente
Emisión	<p>Tiene como objetivo la salida impresa de datos.</p> <ul style="list-style-type: none"> • Son listadores cuando emiten listados como se ingresan sin más que títulos. • Se considera padrón cuando los datos ingresados están ordenados y se emiten además totales finales.
Estadísticos	Recorrido del archivo para contabilizar elementos , utilizando una tabla (memoria interna) y al finalizar emitir un cuadro de resumen .
Corte De Control	<p>Son padrones, pero poseen totales parciales.</p> <p>Es requisito obligatorio que el archivo de entrada esté ordenado por clave compleja.</p>

CORTE DE CONTROL

Características

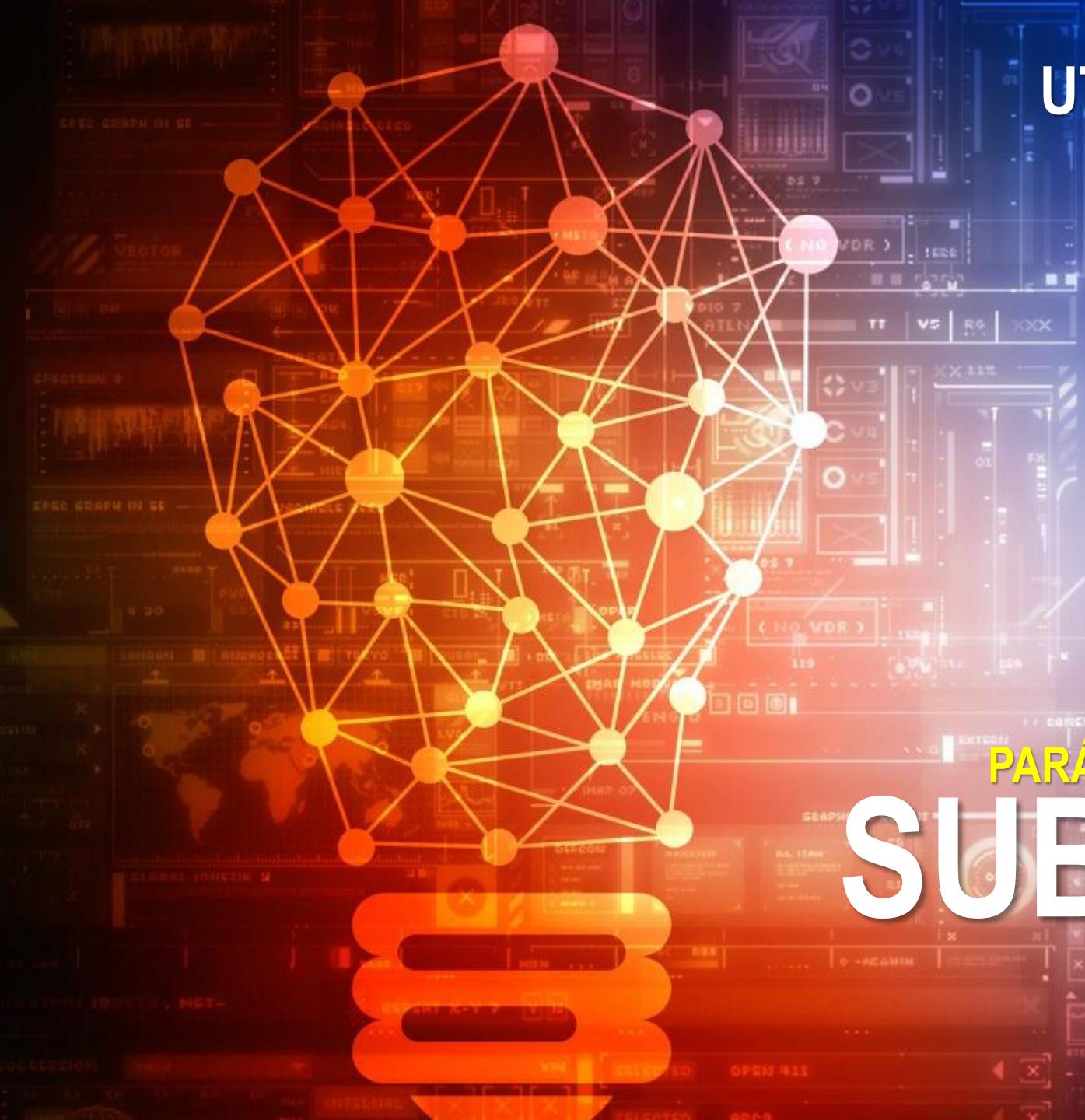
- 1- Proceso de datos de tipo **ESTADISTICO**.
- 2- Proceso **UNITARIO** de archivos con **ACCESO SECUENCIAL**.
- 3- Requiere los datos agrupados de manera jerárquica.
- 4- Reutiliza las variables propias del proceso.

PROCESOS MULTIPLES

Un proceso **es múltiple** cuando existen **mdos o mas Archivos de Entrada** y **1 o ningún Archivo de Salida.**



FUNCIONES, PROCEDIMIENTOS,
PARÁMETROS Y ÁMBITOS DE VARIABLES
SUBACCIONES



ACCION PRINCIPAL

ACCION [Nombre] **(PARÁMETROS)** ES
[AMBIENTE] → Definición de tipos, constantes y variables
(GLOBALES)

[SUB ACCIONES] > Procedimientos y Funciones del algoritmo

ALGORITMO o [PROCESO]

[ACCIONES] → Algoritmo principal

FIN ACCION

SUBACCIONES

SUB-ACCION [Nombre] **(PARÁMETROS)** : **[Tipo_dato_retorno]** ES;

[AMBIENTE] → Definición de tipos, constantes y variables
(LOCALES)

ALGORITMO o **[PROCESO]**

[ACCIONES] → Algoritmo principal de la sub-acción

FIN SUB-ACCION

TIPO DE SUBACCIONES

Obtener el descuento que le corresponde al Cliente

FUNCIONES

PROCEDIMIENTOS

Actualizar el interés acumulado

QUÉ SON LOS PROCEDIMIENTOS

- Un procedimiento es una sub-acción que ejecuta una tarea determinada.
- Son conjuntos de acciones agrupadas que se usan para evitar duplicación de código, conseguir soluciones más legibles y mantenibles.
- Permiten dividir un problema en sub-problemas logrando de esta manera escribir más fácilmente soluciones grandes y complejas.
- Pueden recibir valores de entrada (llamados parámetros).

QUÉ SON LAS FUNCIONES

- (ídem PROCEDIMIENTOS)

Agregamos que:

- Es una sub-acción que retorna un **valor de resultado**.

TIPO DE PARÁMETROS

POR VALOR

POR
REFERENCIA

ÁMBITO DE VARIABLES

Se utilizan
en todo el
algoritmo

GLOBALES

LOCALES

Se utilizan solo
en la subacción
que las define

VARIABLES GLOBALES

Son variables definidas al comienzo del algoritmo (antes de cualquier sub-acción), que se pueden usar a lo largo de todo el algoritmo, es decir, dentro del algoritmo principal y en cada sub-acción definida en el algoritmo.

VARIABLES LOCALES

Son variables definidas dentro de cada sub-acción, y que solo se pueden usar en la sub-acción, en la que son declaradas.